**一、抽象类的定义**   
抽象类是抽象方法和非抽象方法的集合。在特殊情况下，抽象类中全部是抽象方法或者全部是非抽象方法。

【访问权限修饰符】 abstract class 类名{

//代码

}

**二、抽象方法**   
只有抽象类才包含有抽象方法；   
抽象方法的定义与使用；

//抽象方法只是定义方法名称，无方法体，不干具体事的

\*\*定义抽象方法\*\*

[访问权限修饰符] abstract 返回值类型 抽象方法名(参数列表)；

//抽象方法只有派生才能调用，去干事情。

\*\*使用抽象方法\*\*

public class A extends B{

//代码

}

抽象类实例：

//定义抽象类A

abstract class A{

//定义抽象方法method

abstract int method(int a,int b);

}

//定义类C并继承抽象类A

class C extends A {

//派生method方法，实现计算a+b

int method(int a, int b) {

int c=a+b;

System.out.println(c);

return c;

}

}

public class B {

public static void main(String[] args) {

//创建对象并实例化

C c=new C();

c.method(1, 5);

}

}

**三、抽象类特点**   
**1、抽象类不能被实例化。不能直接new对象。   
2、抽象类中可以包含抽象方法（不能有方法体），也可以包含非抽象方法（可以有方法体）。   
3、抽象类不能直接实例化，可以通过其子类实例访问其成员，必须实现抽象类中所有的抽象方法。   
4、抽象方法不能为static、final、protected**

**四、在下列情况下，一个类必须声明为抽象类**   
**1、当一个类的一个或多个方法是抽象方法时；   
2、当类是一个抽象类的子类，并且没有实现父类的所有抽象方法，即只实现部分；   
3、当一个类实现一个接口，并且不能为全部抽象方法都提供实现时。**